IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application No.: 10/726,902	§	Examiner: Fennema, Robert E.
Filed: December 3, 2003	§	Group/Art Unit: 2183
Inventor:	§	Atty. Dkt. No: 5500-88700
Mitchell Alsup, et al.	§	
	§	
	§	
	§	
Title: Transitioning from Instruction Cache to Trace Cache on Label Boundaries	§	
	§	
	§	
	§	
	§	
	§	

REPLY BRIEF

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

This brief is in reply to the Examiner's Answer mailed November 6, 2008. Appellants respectfully request that this Reply Brief be entered pursuant to 37 C.F.R. § 41.41 and considered by the Board of Patent Appeals and Interferences.

REPLY

The Examiner rejected claims 1, 2, 11-15, 24-26, and 32-34 under 35 U.S.C. § 103(a) as being unpatentable over Rotenberg, et al. (hereinafter "Rotenberg") in view of Xia, claims 3 and 16 under 35 U.S.C. § 103(a) as being unpatentable over Rotenberg and Xia in view of Patterson, et al. (hereinafter "Patterson"), claims 4, 10, 17 and 23 under 35 U.S.C. § 103(a) as being unpatentable over Rotenberg and Xia in view of Braught, claims 5-8 and 18-21 under 35 U.S.C. § 103(a) as being unpatentable over Rotenberg, Xia, Braught and further in view of Lange, et al. (U.S. Patent 3,896,419) (hereinafter "Lange"), claims 9 and 22 under 35 U.S.C. § 103(a) as being unpatentable over Rotenberg, Xia and Braught in view of Akkary, et al. (U.S. Patent 6,247,121) (hereinafter "Akkary"), and claims 27-31 and 35 under 35 U.S.C. § 103(a) as being unpatentable over Rotenberg, Xia, Braught and Lange in view of Akkary. Appellants respectfully traverse these rejections for at least the reason given in the Appeal Brief and the following reasons. Different groups of claims are addressed under their respective subheadings. Appellant's arguments from the Appeal Brief filed September 2, 2008 regarding the rejection of these claims are herein incorporated by reference.

First ground of rejection:

The Examiner rejected claims 1, 2, 11-15, 24-26, and 32-34 under 35 U.S.C. § 103(a) as being unpatentable over Rotenberg in view of Xia.

Claims 1, 2, 12, 14, 15, 25, and 32-34:

1. **The cited art clearly fails to teach or suggest** ***wherein the prefetch unit is configured to fetch instructions from the instruction cache until the branch prediction unit outputs a predicted target address, wherein the prefetch unit is configured to check the trace cache for a match for the predicted target address in response to the branch prediction unit outputting the predicted target address, wherein***

***the prefetch unit is configured to not check the trace cache for a match until the branch prediction unit outputs the predicted target address*, as recited in claim 1.**

The Examiner previously submitted that these limitations are inherent in the system of Rotenberg. Appellants have argued that in Rotenberg **every fetch address is compared to the entries in the trace cache, regardless of the operation of the branch prediction unit.** By contrast, Appellants' claim 1 recites that the prefetch unit is configured to check the trace cache for a match for the predicted target address in response to the branch prediction unit outputting a predicted target address, and also recites that the trace cache is not checked for a match until the branch prediction unit outputs the predicted target address. In other words, the trace cache is not checked for a match on every cycle, but only on cycles for which the branch prediction unit outputs a predicted target address.

The Examiner has repeatedly argued that the trace cache as disclosed by Rotenberg cannot search for the predicted target address in the cache if it does not know what the predicted target address is, and that no cache or any other type of hardware can find something when it doesn't know what it is looking for. Appellants have asserted that the Examiner's interpretation of the operation of Rotenberg is **demonstrably incorrect.** There is no need in Rotenberg to have a *predicted* target address in order to check for a match in the trace cache. The predicted target address is not what is being compared to the trace cache in Rotenberg. Instead, Rotenberg explicitly checks for a match on every cycle, regardless of the operation of the branch prediction unit and whether it outputs a predicted target address.

In the Final Action mailed July 18, 2007, the Examiner submitted, "When looking at the entire claim in context, it can be seen that the claim is referring to searching the cache for a match for the predicted target address, and that it cannot search the cache for the predicted target address until the predicted target address is generated" (emphasis Examiner's). **Appellants have asserted that the Examiner is misreading the claim.** The referenced limitation of the claim does not recite "not check the trace cache for the

match" (i.e., referring to the match for the predicted target address of the previous limitation), but instead recites "not check the trace cache for a match" (i.e., any match between the current address and the contents of the trace cache). **The plain language of the claim does not support the Examiner's interpretation. As explicitly recited in claim 1**, multiple instructions are fetched from the instruction cache, **without checking for a match (hit) in the trace cache**, until the branch prediction unit outputs a predicted branch address (i.e., until a branch instruction is encountered for which the branch prediction unit outputs a predicted target address). Only then (**in response to this output**) is the trace cache checked for a match. This functionality is clearly not suggested by any evidence of record.

In his Answer, the Examiner again **misquotes** the above-referenced claim limitation in his remarks this way, "wherein the prefetch unit is configured to not check the trace cache for a match for the predicted target address in response to the branch prediction unit outputting the predicted target address." **Appellants again assert that the Examiner is ignoring the language recited in the claim by improperly reading additional words into the above-referenced limitation that are not recited in the claim itself. As discussed above and in previous Responses, the Examiner's interpretation of this limitation is unsupportable in light of the plain language recited in the claim and in light of Appellants' disclosure.**

In his Answer, the Examiner also states, "Appellant continues to cite the next limitation when referring to the Examiner's argument for this limitation, instead of addressing this limitation by itself." Thus, the Examiner appears to admit that he is attempting a piecemeal reconstruction of Applicants' invention in hindsight without considering the claimed invention as a whole. Such reconstruction is improper. *See, e.g., Interconnect Planning Corp. v. Feil*, 774 F.2d 1132, 1143, 227 USPQ 543, 551 (Fed. Cir. 1985). **Appellants assert that the Examiner has failed to consider the combination of the above-referenced limitations in the claim as a whole.** When the claim is considered as a whole, the cited art clearly fails to teach or suggest the specific features and functionality in the specific manner combined in Appellants' claim. Appellants

again assert that one of ordinary skill in the art having benefit of Appellants' disclosure could not agree with the Examiner's interpretation of the limitation, *"wherein the prefetch unit is configured to not check the trace cache for a match until the branch prediction unit outputs the predicted target address."* The Examiner's repeated assertions that it is impossible to search for something that has not yet been generated simply show a misunderstanding on the Examiner's part of what is described in the cited art and are irrelevant to Appellants' argument and to what is actually recited in the claim. The Examiner's interpretation is also completely inconsistent with Appellants' specification (see, e.g., FIG. 3 and pages 15-18). Appellants further note that the Examiner has previously admitted that Rotenberg teaches that the trace cache is searched on every instruction, which clearly cannot be considered to be in response to the branch prediction unit outputting a predicted target address, as required by claim 1.

In his Answer, the Examiner further states, "Therefore, Examiner believes that the Appellants arguments in Section 1 are moot, as the Appellant appears to be arguing that the Examiners rejection for one limitation does not apply to another, which Examiner has not attempted to argue, and Examiner has already agreed that Rotenberg does not teach the limitation the Appellant argues in Section 1 that Rotenberg does not teach." Appellants note that Section 1 is directed to several limitations (listed above). **Appellants again assert that the Examiner has clearly failed to consider the combination of these limitations as a whole.** In addition, the Examiner appears to be admitting that he is simply relying on each reference individually. This is the hallmark of hindsight reconstruction. It is not proper to pick and chose isolated teachings from references to reconstruct an applicant's claim. Instead, the Examiner must consider the references as a whole, including parts that teach away. M.P.E.P. § 2141.02, last paragraph; *W.L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 220 USPQ 303 (Fed. Cir. 1983), *cert. denied*, 469 U.S. 851 (1984).

For at least the above reasons, the Examiner has failed to establish a *prima facie* rejection. Appellants assert that the combined art does not teach the combination of the

above-referenced limitations arranged as recited in claim 1 when the claim is considered in its entirety.

2. **The cited art clearly fails to teach or suggest *wherein the prefetch unit is configured to not check the trace cache for a match until the branch prediction unit outputs the predicted target address*, as recited in claim 1.**

The Examiner has admitted that Rotenberg fails to teach this limitation and relies on Xia to teach it, citing Section 5.5, first scheme. Specifically, the Examiner stated, "Xia teaches that an advantage of starting a trace only at predictable branch instructions… is that less cache space is required, in direct contrast to Rotenberg, which teaches tracing on all instructions, requiring significantly more space (second scheme)." **Appellants have asserted that Xia's trace table and instruction cache are also checked in parallel on every instruction.** There is nothing in Xia that describes how, or more importantly when, the trace cache is checked for the start of a trace line, or when the system begins a search of the trace cache for any reason. The Examiner's statement made in remarks regarding claim 32 in the Final Action of July 18, 2007, "The advantages that Examiner indicated… are obvious advantages one of ordinary skill in the art would recognize, searching a cache takes time and energy, and if there is no possible way that searching a cache can benefit the user, there is absolutely no reason to do so" **contradict the teachings of both of his own references, which search the trace cache for a hit (although not necessarily a trace start) on every cycle. This is in direct conflict with the limitations of claim 1.**

In his Answer, the Examiner states, "Appellant has completely ignored the Examiners actual rejection, which points to Section 5.5 of Xia, which shows an advantage for only starting traces on branches. Additionally, Appellant has ignored the Examiners rejection where the Examiner has stated that given this information, one of ordinary skill in the art would clearly recognize that if traces can only start on branches, and you do not have a branch instruction, there is absolutely no chance that you could possibly have a hit in the trace cache, and therefore, it would be wasteful to even attempt

to search for something that is guaranteed not to be there." Appellants assert that, as discussed in more detail below, there is nothing in Xia that teaches the improvement suggested by the Examiner (i.e. a change to the method for searching a trace cache), as the only reason stated for starting traces on branches is to save memory space. Such an improvement would not require, nor suggest a change in the method for searching the trace cache, nor does Xia describe an embodiment in which the trace cache is not searched on every cycle. **Appellants again assert that a combination of two references that both teach away from the claimed invention by searching the trace cache for a hit (although not necessarily a trace start) on every cycle, clearly cannot teach or suggest the above-referenced limitation of claim 1.**

In his Answer, the Examiner also states, "Appellant has argued each reference individually, and has attempted to bodily incorporate the two, instead of recognizing that Xia is an improvement over Rotenberg, and applying the references as such." Appellants again assert that the Examiner's stated reason to combine the references is not commensurate with the features he is attempting to combine to result in the claimed invention. The stated advantage of starting traces on branches in Xia (saving memory space) pertains to a different aspect of the system, and not to the feature the Examiner is seeking to combine with Rotenberg (not searching for a match in the trace cache until a predicted target address is output by the branch prediction unit, which is not taught by either reference). Therefore, the Examiner's reason to combine is improper.

For at least the above reasons, the Examiner has failed to establish a *prima facie* rejection.

3. The Examiner has failed to provide a proper reason to combine the references.

The Examiner previously argued that Xia teaches a reason as to why one would only want to start a trace on a branch (smaller cache size), and that one of ordinary skill in the art would be able to take that knowledge, and when incorporating it into

Rotenberg, realize that if traces only start on branches, that it is impossible to get a trace cache hit when the instruction is not a branch. The Examiner has argued that it would be a waste of power to even try (and may also cause critical path problems), issues that those of ordinary skill in the art would clearly be able to recognize, and thus would implement the invention as claimed, because it is an obvious modification based on the information provided by the references.

As discussed above, the Examiner's stated reason for combining the references in teaching the above-referenced limitation, i.e. his assertion that such a technique would contribute to power saving or critical path reduction, is completely unsupported in the cited art. Appellants have argued that the only advantage noted in the references for starting traces on branches is to save memory space. In addition, Appellants asserted that such a change does not require or even suggest a change to the method for searching the trace cache, as the Examiner incorrectly contends. Furthermore, Appellants have argued that there is no evidence of record that changing the search method, as suggested by the Examiner (but not by either of the cited references), would necessarily contribute to power savings or critical path reduction in a system that includes the combination suggested by the Examiner, as this would be dependent on the specific circuitry used to implement such a combination. The Examiner's assertion, therefore, is nothing but pure speculation.

In his Answer, the Examiner submits, "both power and time savings are basic fundamental issues that one of ordinary skill in the art would clearly recognized, as laid out by the Examiner earlier." Appellants' argument was not whether power and time savings are fundamental issues, but that the Examiner's suggested combination of the references would not necessarily result in such savings, as this would be dependent on the specific circuitry used to implement such a combination. Since neither reference teaches the feature *not searching for a match in the trace cache until a predicted target address is output by the branch prediction unit*, it is not at all clear how (or even if) such a medication could be made in a combined system of Rotenberg and Xia, or that the

modification required to implement such a feature would save time or power over the unmodified systems of Rotenberg and/or Xia.

For at least the above reasons, the Examiner has failed to establish a *prima facie* rejection.

4. Even if the references were combined, their combination fails to teach or suggest the claimed invention.

The Examiner has previously stated, "Examiner is using Xia to teach the advantages of starting traces on branches, and is not attempting to bodily incorporate the two references… Examiner reminds Applicant that the test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference; nor is it that the claimed invention must be expressly suggested in any one or all of the references. Rather, the test is what the combined teachings of the references would have suggested to those of ordinary skill in the art." **Appellants have asserted that the Examiner's repeated contention that it would be obvious not to search the trace cache for a non-branch instruction in a system that incorporates the trace-starting feature of Xia is completely unsupported by the cited art, as his own references (including Xia) check for a match on every instruction cycle.** It is unclear how the referenced feature of Xia can possibly teach or suggest the above-referenced limitation of claim 1 in the combined system if it does not teach or suggest this limitation in its own system.

As discussed at length above, Appellants assert that even if the referenced feature of Xia (i.e., starting traces only on branches) were incorporated into the system of Rotenberg, this would not teach or suggest any change to the **search** method of the combined system, much less a change that is not taught or suggested by either reference. **At most, it would result in a system in which traces begin on branches, as described in Xia, but in which the trace cache of Rotenberg (and/or the trace table and instruction cache of Xia) are still checked in parallel on every instruction.**

In his Answer, the Examiner states, "Xia teaches that an optimization of the cache (Section 5.5) is that the trace lines start from branches. The advantage of this, by itself, is that the cache miss rate can be reduced, a significant time saving feature. That is what is expressly suggested in the references." **The Examiner's remarks are completely unsupported in the reference itself.** As discussed above, the only advantage noted in Xia for starting traces on branches is to save memory space. No other advantages are described or suggested, nor would this optimization necessarily result in a reduction in the cache miss rate, much less a "significant time saving feature," as the Examiner contends. **The Examiner's explanations of how missing in the cache incurs great penalty and how this would be avoided in the combined (and modified) system he is suggests are nothing but pure speculation.** In his remarks, the Examiner appears to equate searching the trace cache for a hit and searching the trace cache for a trace start. However, Appellants again assert that both Xia and Rotenberg **search the trace cache for a hit (although not necessarily a trace start) on every cycle.** Therefore, the Examiner's own references teach away from this feature.

For at least the above reasons, the Examiner has failed to establish a *prima facie* rejection. For at least the reasons above, the rejection of claim 1 is unsupported by the cited art and removal thereof is respectfully requested.

Independent claims 14 and 32 include limitations similar to claim 1, and so the arguments presented above apply with equal force to this claim, as well.

Claims 11 and 24:

1. The cited art clearly fails to disclose *wherein each of the plurality of traces comprises partially-decoded instructions*, as recited in claim 11.

The Examiner previously submitted that it is inherent that a trace comprises a partially-decoded instruction, otherwise the necessary control information as shown in

section 2.2 of Rotenberg would not be available. Appellants asserted, however, that the control information described in section 2.2 is control information added to a trace cache entry when the trace cache entry is generated and stored along with the instructions themselves in the trace cache. The control information described in section 2.2 comprises control information generated by the branch predicator and other logic of the trace cache, not information decoded from the instruction itself. **Therefore, the Examiner's assertion that traces inherently comprise partially-decoded instructions is clearly incorrect.** Nothing in Rotenberg describes that the instructions in a trace entry must be partially decoded in order to generate the control information, nor would generating the control information described in section 2.2 necessarily require such decoding.

The Examiner also previously submitted that the traces of Rotenberg hold data that cannot be obtained without some form of decoding, noting, "the traces hold the start and next addresses of the trace, information which is impossible to obtain without decoding, since an instruction is just a random collection of bits until it is decoded and put into a context. The branch flags and branch mask bits also require decoding, since simply identifying that there is even one branch in the trace requires that that branch instruction was decoded to identify it as a branch." Appellants' argument was not that decoding is not ever performed on instructions, but that the instruction traces themselves do not comprise partially decoded instructions. Section 2.2 of Rotenberg states, "The trace cache is made up of instruction traces, control information, and line-fill buffer logic," clearly making a distinction between instruction traces and the control information described above (e.g., the start and next addresses, branch flags, etc.) **This control information is not part of the instruction traces themselves.** The instruction traces store instructions that are output to a mux (shown in FIG. 4), and are not described as included decoded information. In fact, they are depicted as passing to an instruction latch as un-decoded instructions. This is clearly illustrated in FIG. 4 by the fact that the stored instructions pass from the instruction latch to the decoder.

In his Answer, the Examiner repeats his previous assertions and further submits, "Appellant appears to be arbitrarily declaring what is and isn't part of the trace. Examiner asserts that if data is gathered which is required for the trace to work properly, and was gained by decoding the instructions, then it is a partially-decoded part of the instruction trace. The very nature of trace cache requires a partial decoding, and for Appellant to suggest that this is incorrect is a misunderstanding of the art, and incorrect." **Appellants assert, however, that the declaration of what is or is not part of an instruction trace noted above is found in the Examiner's reference itself, as quoted above. Furthermore, Appellants again note that Appellants' argument was not that decoding is not performed on instructions in a system that employs a trace cache, but that in Rotenberg, the instruction traces themselves do not comprise partially decoded instructions, as required by claim 11.**

For at least the reasons above, the rejection of claim 11 is unsupported by the cited art, and removal thereof is respectfully requested.

Claim 24 includes limitations similar to claim 11, and so the arguments presented above apply with equal force to this claim, as well.

Claims 13 and 26:

1. The cited art clearly fails to disclose *wherein each of the plurality of traces is associated with a flow control field comprising a label for an instruction to which control will pass for each branch operation comprised in that trace,* as recited in claim 13.

The Examiner previously submitted that Rotenberg teaches this limitation in Section 2.2, and that the "trace target address" and "trace fall-through address" are labels that describe where control will flow based on each branch operation, based on a prediction. Appellants argued that these addresses are actually described as corresponding to the next fetch addresses if the last branch in the trace is predicted taken

or not taken, respectively. There is nothing in Rotenberg that describes labels for an instruction to which control will pass for each branch operation comprised in the trace, as recited in claim 13, only for the last branch in the trace.

The Examiner later submitted that Section 2.2 of Rotenberg teaches, "the branch flags exist for every branch in the trace, and indicate which instruction control will pass to (the taken or not taken))." **The Examiner is incorrect.** This passage describes that each trace includes a branch flag for each branch within a trace indicating which path was taken by the actual trace (i.e. whether the branch was taken or not taken) for that branch. It does not, however, describe that a label for an instruction to which control will pass is included for each instruction. **Instead, as previously noted, the trace includes a trace target address and a trace fall-through address only for the last branch in the trace.**

In his Answer, the Examiner submits, "However, the branch flags in every single instruction trace indicate in which direction each and every branch in the trace goes. Additionally, given the lack of a solid definition for "label", Examiner asserts that the rejection is proper." Appellants assert that the Examiner's interpretation of the branch flags of Rotenberg as the 'labels' of Appellants claims is completely unsupportable in light of Appellants' specification and in light of the plain language of the claim. For example, the claim recites "*a flow control field comprising a label for an instruction to which control will pass.*" A single bit (binary) branch flag that reflects which of two paths (taken or not taken) was taken following an actual branch, as described in Rotenberg, clearly does not meet this limitation, as it does not specify or indicate a label for a specific instruction to which control will pass.

For at least the reasons above, the rejection of claim 13 is unsupported by the cited art, and removal thereof is respectfully requested.

Claim 26 includes limitations similar to claim 13, and so the arguments presented above apply with equal force to this claim, as well.

Second ground of rejection:

The Examiner rejected claims 3 and 16 under 35 U.S.C. § 103(a) as being unpatentable over Rotenberg and Xia in view of Patterson.

1. The cited art clearly fails to teach or suggest *wherein the branch prediction unit is configured to output the predicted target address in response to detection of a branch misprediction,* as recited in claim 3.

The Examiner has admitted that Rotenberg fails to disclose this limitation and relies on Patterson to teach it. The Examiner has previously submitted that Patterson teaches, "in order to reduce the penalty for a misprediction, you can fetch both the taken and not taken instructions, and put them in the BTB, which would then be able to immediately output the correct path on a misprediction without having to fetch it (Pages 276-277)." Appellants have argued that this passage actually describes reducing the penalty for a misprediction by fetching instructions from both the predicted and unpredicted directions, e.g., by using a dual-ported cache, an interleaved cache, or by fetching from one path and then the other (i.e., fetching from both paths before determining whether a branch has been mispredicted). It does not describe outputting a predicted target address (to be checked against entries in a trace cache, as in Appellants' claims) in response to detection of a branch misprediction.

The Examiner later submitted that one of ordinary skill in the art at the time the invention was made would have stored both the taken and not taken branch paths in the BTB in order to reduce the misprediction penalty, and thus increase performance "by allowing the other path to be immediately output when the misprediction is noted" and "in the combination proposed, the branch predictor holds both the taken and not taken paths, therefore, on a misprediction, the other path is available to be output immediately, thus teaching the limitation." Appellants note that claim 3 does not recite a branch predictor holding both the taken and not taken branch paths so that "the other path is

available to be output immediately", but explicitly recites outputting the predicted target address **in response to** detection of a branch misprediction. There is nothing in the Examiner's citation that describes a mechanism for outputting the predicted target address in response to detection of a branch misprediction.

In his Answer, the Examiner repeats his remarks above, and further states, "Appellant has argued that Claim 3 does not teach a branch predictor." This is clearly not what Appellant has argued, as shown above.

For at least the reasons above, the rejection of claim 3 is unsupported by the cited art, and removal thereof is respectfully requested.

Claim 16 includes limitations similar to claim 3, and so the arguments presented above apply with equal force to this claim, as well.

Third ground of rejection:

The Examiner rejected claims 4, 10, 17 and 23 under 35 U.S.C. § 103(a) as being unpatentable over Rotenberg and Xia in view of Braught.

Claims 4 and 17:

For at least the reasons presented herein regarding the claims from which they depend, the rejection of claims 4 and 17 is unsupported by the cited art, and removal thereof is respectfully requested.

Claims 10 and 23:

1. The cited art clearly fails to teach or suggest *wherein the trace generator is configured to generate traces in response to instructions being decoded,* as recited in claim 10.

The Examiner previously submitted that Rotenberg teaches this limitation in Section 2.2, "the trace can not be completed (written into the cache) until the trace target addresses are calculated, which in turn requires the instructions to be decoded." Appellants have asserted that this does not teach the above-referenced limitation of Appellants' claim, in which traces are generated (not just completed) in response to instructions being decoded. There is nothing in Rotenberg that describes trace generation being performed in response to instructions being decoded. The Examiner has also previously submitted that the trace requires decoded instructions in order to be created. Appellants have argued that the cited art does not teach that traces comprise partially decoded instructions. In addition, Appellants have argued that even if instructions included in traces are decoded at some point, there is nothing in the Examiner's citations that teaches or suggest **generating traces in response to this decoding**, as required by claim 10.

In his Answer, the Examiner submits, "the trace requires data from decoded instructions in order to make the trace, thus, the traces are generated in response to the decoding, even if it is not an immediate response." Appellants again assert that even if instructions are decoded in Rotenberg and the operation of the trace cache depends in some way on this decoding, the Examiner's remarks implying a cause-and-effect relationship between decoding instructions and generating traces (as required by Appellants' claim) are clearly unsupported in the references themselves.

2. The Examiner has not provided any motivation or other reason to combine the references in teaching the specific limitations of claims 10. Therefore,

the Examiner has failed to establish a *prima facie* obviousness of the claimed invention.

For at least the reasons above, the rejection of claim 10 is unsupported by the cited art, and removal thereof is respectfully requested.

Claim 23 includes limitations similar to claim 10, and so the arguments presented above apply with equal force to this claim, as well.

Fourth ground of rejection:

The Examiner rejected claims 5-8 and 18-21 under 35 U.S.C. § 103(a) as being unpatentable over Rotenberg, Xia, Braught and further in view of Lange.

Claims 5 and 18:

1. The cited art clearly fails to teach or suggest *wherein the trace generator is configured to check the trace cache for a duplicate copy of the trace that the trace generator is constructing,* as recited in claim 5.

The Examiner has previously admitted that Rotenberg, Xia, and Braught fail to teach this limitation and relied on Lange to teach it. As discussed below regarding claims 27 and 35, Lange's description of the operation of a data cache during fetching from main memory has absolutely nothing to do with the specific limitations of claim 5. Appellants assert that identifying a duplicate copy of a value in a cache when merely fetching a value from memory is clearly not analogous to identifying a trace entry corresponding to a trace entry under construction (e.g., in response to decoding and/or execution of previously fetched instructions), as in Appellants' claimed invention. Nothing in Lange teaches or suggests a mechanism for the identification of such trace entries.

Appellants have argued that checking Rotenberg's trace cache for a duplicate trace before collecting a new trace is also contradictory to the Examiner's remarks regarding the advantages of including duplicate traces. **The Examiner first argues that Rotenberg teaches duplicate traces may exist, and then argues that the combined references teach that the system of Rotenberg should not allow construction of duplicate traces. These arguments cannot coexist if the references are to teach the limitations of claim 5. If no duplicate traces can be constructed, there would be no reason to check the trace cache for a duplicate copy of a trace that the trace generator is constructing.**

In the Response to Arguments section of the Office Action of March 31, 2008, the Examiner states that Applicant has argued that the references do not teach the limitations of claim 5 and that Applicant has argued that Rotenberg does not allow for duplicate traces, thus cannot claim them. **The Examiner has mischaracterized Appellants' argument above. Appellants' argument above is that the Examiner's remarks regarding the teaching of claim 5 contradict each other,** as discussed in more detail below, and therefore do not establish a *prima facie* obviousness of the claimed invention.

In his Answer, the Examiner repeats his incorrect assertion that Appellant has ignored the possibility of having duplicate traces. **As explained above, it is the Examiner who has attempted to combine two alternate embodiments that cannot coexist.**

2. The Examiner has not provided a valid reason to combine the references.

The Examiner submits that one of ordinary skill in the art at the time the invention was made would have been motivated to combine the teachings of Lange's checking for duplicates and discarding of the duplicates to Rotenberg in order to solve either or both of the problems of getting a miss while servicing a miss (in a system that allows duplicate traces), and preventing eviction of a useful trace by a duplicate trace (in a system that

does not allow duplicate traces). These are clearly contradictory goals with contradictory solutions. It is not clear how the first goal (i.e. handling getting a cache miss while servicing a miss) could solved by checking for duplicate traces and discarding duplicates in a system that allows duplicate tracing, as suggested by the Examiner. **First, handling a cache miss while servicing another cache miss has nothing to do with the limitations of claim 5, which is directed to trace generation.** Furthermore, if the solution to this problem involved detecting duplicate traces during construction and then discarding them, the system would, in fact, not allow duplicate traces, in contrast to the Examiner's suggestion. Thus, it would not be consistent with the "judicial trace selection" example described below in remarks regarding claim 27. In addition, as discussed above, the referenced feature of Lange does not describe checking for a duplicate copy of a trace while constructing a trace, as it has nothing to do with the operation of a trace cache at all. Therefore, it cannot be used to solve the second problem cited by the Examiner (preventing eviction of a useful trace by a duplicate trace).

The Examiner's reasons for combining the references are clearly not supported by the cited art, and the references, when combined, do not teach the limitation recited in claim 5.

For at least the reasons above, the rejection of claim 5 is unsupported by the cited art, and removal thereof is respectfully requested.

Claim 18 includes limitations similar to claim 5, and so the arguments presented above apply with equal force to this claim, as well.

Claims 6 and 19:

1. **The cited art clearly fails to teach or suggest *wherein in response to the trace generator identifying a duplicate copy of the trace, the trace generator is configured to discard the trace under construction,* as recited in claim 6.**

The Examiner previously cited Lange as teaching this limitation in column 5, lines 5-10. Appellants have argued that, as discussed herein, Lange is not directed to trace generation at all. Lange's description of the operation of a data cache during fetching from main memory has absolutely nothing to do with the specific limitations of Appellants' claims. For example, there is nothing in the combination of cited references that teaches or suggests identifying a duplicate copy of a trace and in response to such identification, discarding a trace under construction.

2. The Examiner has not provided any motivation or other reason to combine the references in teaching the specific limitations of claim 6. Therefore, the Examiner has failed to establish a *prima facie* obviousness of the claimed invention.

Appellants note that the Examiner did not provide any additional remarks regarding claim 6 in his Answer.

For at least the reasons above, the rejection of claim 6 is unsupported by the cited art, and removal thereof is respectfully requested.

Claim 19 includes limitations similar to claim 6, and so the arguments presented above apply with equal force to this claim, as well.

Claims 7 and 20:

1. The cited art clearly fails to teach or suggest *wherein in response to the trace generator identifying an entry corresponding to a duplicate copy of the trace, the trace generator is configured to check the trace cache for an entry corresponding to a next trace to be generated,* as recited in claim 7.

The Examiner previously asserted that Rotenberg teaches this limitation in Section 2.2, "The trace cache is checked every time there is a potential new trace, so

when one trace is found and discarded, the next potential new trace will cause the trace cache to be checked again." **Appellants asserted that this clearly does not teach the specific limitations of Appellants' claim, as Rotenberg does not perform the checking in response to identifying a duplicate copy of the trace, as required by claim 7.** Instead, "the trace cache is checked every time there is a potential new trace" regardless of whether a duplicate copy has been identified.

2. The Examiner has not provided any motivation or other reason to combine the references in teaching the specific limitations of claim 7. Therefore, the Examiner has failed to establish a *prima facie* obviousness of the claimed invention.

Appellants note that the Examiner did not provide any additional remarks regarding claim 7 in his Answer.

For at least the reasons above, the rejection of claim 7 is unsupported by the cited art, and removal thereof is respectfully requested.

Claim 20 includes limitations similar to claim 7, and so the arguments presented above apply with equal force to this claim, as well.

Claims 8 and 21:

1. The cited art clearly fails to teach or suggest *wherein in response to the trace generator identifying a trace entry corresponding to the next trace to be generated, the trace generator is configured to discard the trace under construction*, as recited in claim 8.

The Examiner previously submitted that Lange teaches this limitation in column 5, lines 5-10. **Appellants have argued that, as discussed above, Lange teaches nothing about the generation of trace entries, much less the specific limitations of**

claim 8. For example, there is nothing in the combination of references that teaches or suggests discarding a trace **under construction** in response to identifying a trace entry corresponding to the next trace to be generated (i.e., a trace other than the current trace entry).

The Examiner later submitted, "Lange teaches an analogous situation to the claim, where a duplicate copy of a value in a cache is discarded when it is found to be a duplicate, which is what the claim is claiming." Appellants again asserted that identifying "a duplicate copy of a value in a cache" when merely fetching a value from memory is clearly not analogous to identifying a trace entry corresponding to a trace entry under construction (e.g., in response to decoding and/or execution of previously fetched instructions) much less to identifying the next trace to be generated, during construction of a current trace (from executed instructions). Nothing in Lange teaches or suggests a mechanism for the identification of such trace entries.

In his Answer, the Examiner states, "Lange teaches that there are extreme disadvantages to storing duplicate trace copies, therefore, if one is discovered, one of ordinary skill in the art would clearly be motivated to discard the duplicate trace if it is discovered." Appellants again assert that Lange teaches nothing about traces, much less about discovering or storing duplicate trace copies, as suggested by the Examiner.

2. The Examiner has not provided any motivation or other reason to combine the references in teaching the specific limitations of claim 8. Therefore, the Examiner has failed to establish a *prima facie* obviousness of the claimed invention.

The Examiner previously submitted, "Additionally, Examiner has already laid out a motivation for combining Lange with Rotenberg, and Examiner refers the Applicant to the rejection to see said motivation." However, as discussed above, the referenced feature of Lange is not analogous to the limitations recited in claim 8, or in others of the dependent claims.

For at least the reasons above, the rejection of claim 8 is unsupported by the cited art, and removal thereof is respectfully requested.

Claim 21 includes limitations similar to claim 8, and so the arguments presented above apply with equal force to this claim, as well.

Fifth ground of rejection:

The Examiner rejected claims 9 and 22 under 35 U.S.C. § 103(a) as being unpatentable over Rotenberg, Xia and Braught in view of Akkary.

1. The cited art clearly fails to teach or suggest *wherein the trace generator is configured to generate traces in response to instructions being retired,* as recited in claim 9.

The Examiner previously admitted that Rotenberg does not teach that instructions need to be retired before the trace can be generated and relies on Akkary to teach this limitation. The Examiner submitted that Akkary teaches that instructions are not put into the trace buffers until they have been retired, to ensure that they executed correctly (column 3, lines 40-44). **This passage actually says just the opposite**, that is, that instructions placed in the trace buffer stay in the trace buffer until they are retired.

The Examiner later submitted, "Applicant is attempting to bodily incorporate the references, as opposed to looking at the rejection that the Examiner has laid out. Akkary teaches that retired instructions are guaranteed to be correct, and Rotenberg teaches that putting incorrect instructions into the trace cache can cause significant problems, therefore the Examiner has said it would have been obvious to one of ordinary skill in the art to wait until an instruction is retired in order to place it in the trace cache, and the manner in which Akkary uses a trace buffer is completely unrelated to the rejection, and

it is improper to bodily incorporate the references as the Applicant is attempting to do to argue the rejection of the claim." **However, Appellants argument above was directed to the Examiner's own remarks**. The Examiner also submitted, "Akkary teaches that instructions are not put into the trace buffers until they have been retired, to ensure that they executed correctly." **This is incorrect**. **Akkary teaches putting instructions in the trace buffers before retirement (i.e., before they are known to be correct).** Thus, Akkary teaches away from the limitations of Appellants' claim 9. Appellants again remind the Examiner, "It is improper to combine references where the references teach away from their combination." *In re Grasselli*, 218 USPQ 769, 779 (Fed. Cir. 1983).

Appellants further argued that the Examiner appears to be relying on each reference individually. This is the hallmark of hindsight reconstruction. It is not proper to pick and chose isolated teachings from references to reconstruct an applicant's claim. Instead, the Examiner must consider the references as a whole, **including parts that teach away.** M.P.E.P. § 2141.02, last paragraph; *W.L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 220 USPQ 303 (Fed. Cir. 1983), *cert. denied*, 469 U.S. 851 (1984). **Since Akkary explicitly teaches that instructions are put into the trace buffers before retirement, Akkary cannot be properly combined with Rotenberg to teach *wherein the trace generator is configured to generate traces in response to instructions being retired.***

In his Answer, the Examiner again asserts that Appellants are attempting bodily incorporation, instead of reading what is taught to one of ordinary skill in the art. The Examiner submits, "The combination of references teaches that retired instructions are guaranteed to be correct, and that making a trace of incorrect instructions would cause useful traces to be evicted from the cache. Examiner believes that one of ordinary skill in the art, knowing these two pieces of information, would clearly see that a trace of retired instructions will be less likely to displace a useful cache line, and therefore, clearly be advantageous." Appellants assert, however, that in remarks regarding claim 9, the Examiner previously referenced Rotenberg, Section 2.3, as teaching that "some traces are committed but never used." The Examiner appears to be implying that these traces are

traces of incorrect instructions. **This is incorrect.** This passage actually describes recent traces that are never reused, and has nothing to do with whether they were correctly executed. Therefore, the combination of references does not teach or suggest the feature suggested by the Examiner.

For at least the reasons above, the rejection of claim 9 is unsupported by the cited art, and removal thereof is respectfully requested.

Claim 22 includes limitations similar to claim 9, and so the arguments presented above apply with equal force to this claim, as well.

Sixth ground of rejection:

The Examiner rejected claims 27-31 and 35 under 35 U.S.C. § 103(a) as being unpatentable over Rotenberg, Xia, Braught and Lange in view of Akkary.

Independent Claims 27 and 35:

1. The cited art clearly fails to teach or suggest *receiving a retired instruction*.

The Examiner previously admitted that Rotenberg does not teach that instructions need to be retired before the trace can be generated and relies on Akkary to teach this limitation. The Examiner submitted that Akkary teaches that instructions are not put into the trace buffers until they have been retired, to ensure that they executed correctly (column 3, lines 40-44). **However, as discussed above, this passage actually says just the opposite**, that is, that instructions placed in the trace buffer stay in the trace buffer until they are retired. Therefore, Akkary teaches away from the limitation above.

2. The cited references fail to teach or suggest *in response to determining that a previous trace under construction duplicates a trace in a trace cache and that the received instruction corresponds to a branch label, beginning construction of a new trace.*

The Examiner admitted that Rotenberg fails to teach this limitation or to discuss the issue of duplication. The Examiner submitted that Rotenberg discusses the disadvantages which occur when a trace cache miss occurs while servicing a previous trace cache miss, and teaches the disadvantages of a useless trace displacing a useful trace. The Examiner then submitted, "Therefore, Rotenberg teaches a system in which allows tracing of potential duplicate traces, and also a system which requires action when a miss occurs while servicing a miss." Appellants asserted that the Examiner is contradicting himself and that he has cited nothing in Rotenberg that teaches tracing of potential duplicate traces. In Rotenberg, the fill-line buffer logic services trace cache misses (i.e., the combination of a matching target address and matching branch flags is not found in the trace cache). There is no reason to believe there would ever be a duplicate trace in the system of Rotenberg, and no reasons or opportunity to avoid such duplication. The Examiner suggested that there is a "potential for duplicate traces to exist with path associativity in Rotenberg's alternative embodiments." **This is completely unsupported in Rotenberg and does nothing to overcome the deficiencies of the cited references in teaching the above-referenced limitations. Similarly, the Examiner's contention that "Rotenberg indicates in his judicial trace selection that storing a duplicate trace would be at best useless, and at worst displace a useful trace that may be used" is also completely unsupported.** The solution discussed in this section is completely different from the specific limitations recited in claim 27. The Examiner submitted that this section provides motivation to handle cases involving duplicate traces, and that Lange provides such a method, by simply discarding the duplicate value. **Again, the Examiner's remarks do not address the limitations recited in claim 27, which are directed to specific conditions to be met before beginning construction of a new trace, and which the cited references do not teach.**

3. The Examiner has failed to provide a proper reason to combine the references.

The Examiner has argued that Akkary suggests the limitation of using a retired instruction, and he includes remarks regarding "correctness" of a trace, a goal of Rotenberg's trace cache to exploit code reuse, and that branches tend to be biased in one direction. However, Rotenberg explicitly states that the trace line is filled as instructions are fetched from the instruction cache, **which is clearly incompatible with filling the trace line with retired instructions**. The fact that the system of Rotenberg could solve this problem in a manner different from the only example described does not suggest the specific solution recited in the claims. **Appellants also asserted that combining the references does not teach the claimed invention,** as neither reference teaches a solution that involves receiving a retired instruction, and in response to determining that a previous trace under construction duplicates a trace in a trace cache and that the received instruction corresponds to a branch label, **beginning construction of a new trace.**

The Examiner previously cited Lange's description of the operation of a data cache during fetching from main memory. Appellants asserted that this has absolutely nothing to do with the specific limitations of claim 27. Checking Rotenberg's trace cache for a duplicate trace before collecting a new trace is also contradictory to the Examiner's remarks above regarding the advantages of including duplicate traces. **The Examiner first argued that Rotenberg teaches duplicate traces may exist, and then argued that the combined references teach that the system of Rotenberg should not allow construction of duplicate traces. These arguments cannot coexist if the references are to teach the limitations of claim 27. If no duplicate traces can be constructed, there would be no reason to determine if a trace previously under construction was duplicating a trace already in the trace cache.** The Examiner's reasons for combining the references are clearly not supported by the cited art, and the references, when combined, do not teach the limitations recited in claim 27.

Appellants note that the Examiner did not provide any additional remarks regarding claim 27 in his Answer. Note also that claim 27 includes limitations similar to those of claim 5 discussed above; therefore the arguments presented above regarding claim 5 apply with equal force to this claim, as well.

For at least the reasons above, the rejection of claim 27 is unsupported by the cited art and removal thereof is respectfully requested.

Claim 35 includes limitations similar to claim 27, and so the arguments presented above apply with equal force to this claim, as well.

Claim 28:

1. The cited art clearly fails to teach or suggest *continuing construction of an incomplete trace already in process in response to determining that the incomplete trace does not duplicate a trace in a trace cache.*

The Examiner previously asserted that Rotenberg teaches this limitation in Section 2.2. Appellants have argued that, as discussed above, the combination of cited references fails to teach or suggest checking the trace cache for a duplicate trace, much less the specific limitations recited in claim 28. In addition, the Examiner has not provided any arguments to support his assertion that this limitation is taught by Rotenberg.

2. The Examiner has not provided any motivation or other reason to combine the references in teaching the specific limitations of claim 28. Therefore, the Examiner has failed to establish a *prima facie* obviousness of the claimed invention.

Appellants note that the Examiner did not provide any additional remarks regarding claim 28 in his Answer.

For at least the reasons above, the rejection of claim 28 is unsupported by the cited art, and removal thereof is respectfully requested.

Claim 29:

1. The cited art clearly fails to teach or suggest *searching the trace cache for duplicate entries subsequent to completion of the previous trace under construction or the new trace.*

The Examiner previously submitted that Lange teaches this limitation in column 5, lines 5-10. **Appellants have argued that, as discussed above, Lange teaches nothing about the operation of a trace cache, much less the specific limitations of claim 29.** For example, there is nothing in the combination of cited references that teaches or suggests searching a trace cache for duplicate entries subsequent to completion of the previous trace under construction (i.e., that of claim 27) or the new trace. In addition, the Examiner has not provided any arguments to support his assertion that this limitation is taught by Lange.

2. The Examiner has not provided any motivation or other reason to combine the references in teaching the specific limitations of claim 29. Therefore, the Examiner has failed to establish a *prima facie* obviousness of the claimed invention.

Appellants note that the Examiner did not provide any additional remarks regarding claim 29 in his Answer.

For at least the reasons above, the rejection of claim 29 is unsupported by the cited art, and removal thereof is respectfully requested.

Claim 30:

1. The cited art clearly fails to teach or suggest *creating a new entry in the trace cache in response to no duplicate entry being identified.*

The Examiner previously asserted that Rotenberg teaches this limitation in Section 2.2. Appellants have argued that, as discussed above, the combination of cited references fails to teach or suggest checking the trace cache for a duplicate trace, much less the specific limitations recited in claim 28. In addition, the Examiner has not provided any arguments to support his assertion that this limitation is taught by Rotenberg.

2. The Examiner has not provided any motivation or other reason to combine the references in teaching the specific limitations of claim 30. Therefore, the Examiner has failed to establish a *prima facie* obviousness of the claimed invention.

Appellants note that the Examiner did not provide any additional remarks regarding claim 30 in his Answer.

For at least the reasons above, the rejection of claim 30 is unsupported by the cited art, and removal thereof is respectfully requested.

Claim 31:

1. The cited art clearly fails to teach or suggest *discarding a trace in response to a duplicate entry being identified.*

The Examiner previously submitted that Lange teaches this limitation in column 5, lines 5-10. **Appellants have argued that, as discussed above, Lange teaches nothing about the operation of a trace cache, much less the specific limitations of**

claim 31. For example, there is nothing in the combination of cited references that teaches or suggests discarding a trace in response to a duplicate (trace cache) entry being identified. In addition, the Examiner has not provided any arguments to support his assertion that this limitation is taught by Lange.

2. The Examiner has not provided any motivation or other reason to combine the references in teaching the specific limitations of claim 31. Therefore, the Examiner has failed to establish a *prima facie* obviousness of the claimed invention.

Appellants note that the Examiner did not provide any additional remarks regarding claim 31 in his Answer.

For at least the reasons above, the rejection of claim 31 is unsupported by the cited art, and removal thereof is respectfully requested.

CONCLUSION

For the foregoing reasons, it is submitted that the Examiner's rejection of claims 1-35 is erroneous, and reversal of his decision is respectfully requested.

The Commissioner is authorized to charge any fees that may be due to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5500-88700/RCK.

Respectfully submitted,

/Robert C. Kowert/
Robert C. Kowert, Reg. #39,255
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: January 6, 2009